REMARKS/ARGUMENTS

Status of the Application

Prior to entry of this amendment, claims 1-29 were pending in the application. A final office action mailed November 3, 2005 rejected claims 1, 3-6, 9-14, 18-22, and 24-29 under 35 U.S.C. § 102(b) as being anticipated by USP 5,416,903 ("Malcolm"). The final office action also rejected claims 2, 7, 8, 15-17 and 23 under 35 U.S.C. § 103(a) as being unpatentable over Malcolm in view of USP 6,526,426 ("Lakritz"). This amendment neither amends, cancels nor adds any claims. Hence, after entry of this amendment, claims 1-29 will remain pending for examination. Reconsideration of the pending claims is respectfully requested.

§ 102 Rejections

The office action rejected claims 1, 3-6, 9-14, 18-22, and 24-29 under § 102(b) as being anticipated by Malcolm. This rejection is respectfully traversed, at least because Malcolm fails to teach or suggest each element of any pending claim. More specifically, Malcolm fails to teach or suggest each element of independent claims 1, 14, 18 and 25, and those claims therefore are believed to be allowable over Malcolm. Furthermore, claims 2-6, 9-13, 19-22, 24 and 26-29 are believed to be allowable at least because they depend from allowable base claims.

Consider, for example, claim 1, which recites, inter alia, "said at least a first record containing source text in a first natural language is displayed to a translator as said first record containing source text in a first natural language will be seen by a user of said computer program." Malcolm fails to teach or suggest at least this element of claim 1. The office action speculates that Malcolm provides for this element "because Malcolm generates subscreens, subwindows or subpanels dynamically, [so] a translator is presented with displays of both pre-release versions and final versions of any program or text." Office Action, at 10.

It is respectfully submitted, however that, this assertion is not supported by Malcolm's disclosure. Specifically, nothing in Malcolm teaches or suggests that the translator is provided with a working version of the application to be translated.

Instead, Malcolm teaches a system in which

> "[a] compiler is designed . . . to segregate language independent and language dependent information. . . . The language dependent information 70 contains text and/or data which is presented to the user in a particular language which, hopefully, the user understands. Thus, this information is segregated in a file separate from the language independent information, so that only the information requiring translation from one language to another need be conveyed to translators or translation centers. . . . .
>
> The translation of the language dependent file 70 results in a corresponding file 80 having prompt strings for the particular language being supported. The translation step 102 merely involves a translator reading the language dependent file 70, and translating the prompts into the desired language, creating a file 100 containing these translated prompts 102.

Malcolm, c. 5, l. 55 – c. 6, l. 31 (emphasis added).

Clearly, Malcolm contemplates that the translators are to be provided only with the language-dependent files, not with the language-independent files, which one skilled in the art would recognize must be required to operate the application being translated. For instance, Malcolm teaches that

> [t]he language independent file 68, and the appropriate language dependent file 70, 100 (or any other language file for a supported language) then serve as inputs to the panel-formatter subsystem 108. The panel formatter interacts with the application program 100 which is running on the data processing system through an API . . . . The panel formatter subsystem 108 renders the screen display, including panels such as 88 of Fig. 5, as directed by the Application program 110, using the language independent file 68 and the appropriate language dependent file 70.

Malcolm, c. 6, ll. 38-51. Unquestionably, if the translator is provided only with the language dependent file (as taught by Malcolm in the text quoted above), the translator would not have the necessary components to run the application and thereby render the "subscreens, subwindows or subpanels dynamically," as suggested by the office action.

Along the same lines, the applicants respectfully submit that the disclosure of Malcolm fails to support the office action's assertions that "Malcolm discloses dynamic generation of subscreens to a person responsible for translating a language specific portion into a subsequent language," Office Action at 10-11, and that "Malcolm must necessarily display a program or text to a translator as it will appear in its final form to an end user." *Id.* at 11. The cited portions of Malcolm (c. 2, l. 55 – c. 3, l. 1, c. 4, ll. 58-63) do not support these assertions.

Instead, the applicants respectfully submit that the office action's interpretation of Malcolm conflates the operation of the application (for a user) from the development process (in which language dependent files are translated). For example, Malcolm does teach that

> [t]he present invention solves the aforementioned problems by providing a system and method for improved user interface screen generation for an application program which supports multilingual users. Dynamic generation of subscreens, or subwindows/subpanels, which conform to a given country's language requirements is provided. These subwindows support both input and output operations between an application program and user of a data processing system.

Malcolm, c. 2, ll. 55-63 (emphasis added). However, it is clear from the quoted text that Malcolm here describes the operation of the application for a user of the application—nothing in Malcolm teaches or suggests that the subscreens or subwindows/subpanels are provided to a translator during the development process, as the office action interprets.

In describing the facilities provided to a translator, the passage of Malcolm cited by the office action merely notes that,

> [t]he system and method further provide improved conveyance of information between the person(s) responsible for the initial user interface screen or panel layout in the initial language, and the person(s) responsible for translating the language specific portion of this screen layout into a subsequent language. Information containing specific comments pertaining to a given field or submenu to be displayed, and its associated text, can be appended to the file containing the text to be displayed. Such annotated text can direct the translator that a particular text string should not be translated, for example, because it is an acronym not capable of being translated. Other information conveyed with the text file can include change log information, which specifies that only a portion of the whole file has been changed, and which portion needs to be translated as a result.

Id., c. 2, l. 64 – c. 3, l. 1 (emphasis added). Clearly, this passage of Malcolm teaches not that the application itself is provided to the translator, but that a text file (and associated information, including annotation text and a change log) is provided to the translator. As noted above, this text file could not be used, without the application itself, to generate any subwindows or panels.

The other passage relied upon by the office action (c. 4, ll. 55-63) merely describes the operation (at run time) of an application developed by Malcolm's system. It contains no teaching or suggestion that Malcolm's system operates this way during the translation phase of development. Nor do the application panels of Fig. 5 provide an indication

of what is provided to the translator. Instead, Fig. 5 illustrates sample panels 88, 120 that would be provided to a user (not a translator) in different languages, according to the panel definition 80. Malcolm provides no teaching or suggestion that the panels of Fig. 5 are displayed to a translator.

Instead, as noted above, Malcolm expressly teaches that only the language dependent file is provided to a translator. An example of this file is the file 70 illustrated by Fig. 4 of Malcolm. Even a cursory review of Fig. 4 establishes that the file 70 is not a "record containing source text in a first natural language [that] is displayed to a translator as said first record containing source text in a first natural language will be seen by a user of said computer program," as recited by claim 1, since the file 70 is completely different in appearance from the panels 88, 120 illustrated by Fig. 5 (which are provided to a user). Instead, the file 70 includes merely plain text, along with labels and an annotation tag (e.g., 134), which "is recognized by the compiler 31 during compilation, and is included in the language dependent file as a comment 134 associated with that particular prompt string 74." Malcolm, c. 9, ll. 34-36. As Malcolm explains, such comments are "useful for aiding translators who will be translating the language dependent file," *id.*, c. 9, ll. 55-57, precisely because Malcolm does not provide to the translator a record containing source text in a first natural language as it will be seen by a user, as recited by claim 1. As Malcolm notes,

> [f]or instance, some words in English may be used as either a noun or a verb, depending on the context of the word in a sentence. When the word is conveyed in the language dependent file, it may not have any context associated with it, and the translated word in the target language may have differing translations depending upon whether it is a noun or verb. By adding this annotation feature, such information can be easily conveyed to translators and translation houses.

Id., c. 9, ll. 58-67 (emphasis added). If Malcolm did in fact display the source text to the translator as it would be seen by a user, the source text would have the necessary context to enable proper translation, and such annotation tags would be unnecessary.

Hence, Malcolm operates in a fundamentally different way than the method of claim 1. Instead of displaying to a translator a record containing source text as that record will be seen by a user of the program, Malcolm provides plain text (with annotation comments) to the

translator, and the annotation comments provide any necessary context. (It should be noted, of course, that the annotation comments of Malcolm themselves render the text displayed to the translator different than the text that will be seen by a user.) For at least this reason, Malcolm fails to teach or suggest each element of claim 1, and claim 1 therefore is believed to be allowable over Malcolm.

Independent claims 14 recites, inter alia, "at least a first portion of text written in said first natural language and said at least a first portion of text into said second natural language are displayed to a translator in a context that would be seen by an end user of said computer program." As noted above, Malcolm does not teach or suggest this feature. Nor does Malcolm teach or suggest any edit boxes, let alone "a translation tool worksheet" comprising "a first edit box, wherein at least a first of said portions of text written in said first natural language and included in said current version of said computer program" and "a second edit box, wherein a translation of said at least a first of said portions of text written in said first natural language and displayed in said first edit box into a second natural language can be entered," as recited by claim 14. As noted above, Malcolm contemplates a translator receiving and editing a text file containing the language to be translated, and Malcolm does not teach or suggest any particular apparatus for performing this editing. Necessarily, then, Malcolm fails to teach software comprising the translation tool work sheet recited by claim 14. For at least these reasons, claim 14 is believed to be allowable over Malcolm.

Independent claim 18 is directed to a "[a] method for developing multiple natural language versions of software," and it recites,, inter alia, "displaying said at least a first component of said software translated into said second language in the context of said first component's usage in said target software program." As noted above, Malcolm specifically teaches that text to be translated is not provided to the translator without context, and that the translator merely edits a text file, with assistance from annotation comments, which are provided precisely because the text to be translated is provided to the translator without context. Hence,

for at least this reason, Malcolm fails to teach or suggest each element of claim 18, and claim 18 therefore is believed to be allowable over Malcolm.

Independent claim 25 recites "translating said first textual content into a second language, wherein said translated first textual content is displayed in said first format, and wherein said first format is the format in which said first textual content will appear to an end user of said computer software." As noted above, Malcolm fails to display textual content in a format in which the content will appear to an end user, and for at least this reason, claim 25 is believed to be allowable over Malcolm.

Accordingly, independent claims 1, 14, 18 and 25 are believed to be allowable over Malcolm, and reconsideration of those claims is respectfully requested. Furthermore, dependent claims 2-6, 9-13, 19-22, 24 and 26-29 are believed to be allowable at least because they depend from allowable base claims, and in the interest of brevity, these dependent claims will not be discussed individually. Reconsideration of claims 2-6, 9-13, 19-22, 24 and 26-29 is also respectfully requested.

§ 103 Rejections

The office action rejected claims 2, 7, 8, 15-17, and 23 under § 103(a) as being unpatentable over Malcolm in view of Lakritz. This rejection is respectfully traversed. Each of the claims rejected under § 103(a) depends from either claim 1, 14 or 18, and as noted above, Malcolm fails to teach each element of these base claims. Lakritz fails to provide the disclosure missing from Malcolm, and claims 2, 7, 8, 15-17, and 23 are believed to be allowable at least because they depend from allowable base claims, so in the interest of brevity, those claims will not be discussed individually. Reconsideration of the claims rejected under § 103(a) is respectfully requested.

Conclusion

In view of the foregoing, the applicant believes all claims now pending in this Application are in condition for allowance. The issuance of a formal Notice of Allowance at an early date is respectfully requested.

If the Examiner believes a telephone conference would expedite prosecution of this application, please telephone the undersigned at 303-571-4000.

Respectfully submitted,

Chad E. King
Reg. No. 44,187

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, Eighth Floor
San Francisco, California 94111-3834
Tel: 303-571-4000
Fax: 415-576-0300
Attachments
CEK:fcr
60656164 v1